

Changes - razord

1.12 (not yet released)

- Wrote razord-sync(1) and razord-alive(1). razord-sync rsyncs the catalogue at bootup and bootstrap. razord-alive broadcasts an ``I am alive" message to all servers in the razor-echo zone at bootup.
- Wrote Razor::Config::findconf() and Razor::Config::findbin() to locate Razor config files and binaries on the system. By default, razord(1) is setup to run from \$ENV{HOME}.
- Razor's various distributed functions are now split accross 4 zones. razor.vipul.net lists the servers that contain an up-to-date copy of the catalogue. razor-check(1) picks up the closest server from this zone to check incoming mail. razor-report.vipul.net lists the servers that accept spam reports for the razor.vipul.net zone. razor-report(1) will send reports to the closest server from this zone. Servers in this zone will echo reports to all catalogue servers listed in the razor-echo.vipul.net zone. Finally, razor-rsync.vipul.net lists servers that provide rsync access to their catalogues for other catalogues to sync with.
- Added `noecho' and `norecordchecks' parameters to razord.conf.
- Added `bind' and `group' parameters to razord.conf and made them mandatory.
- Introduced ECHO protocol. Reports are echoed to other razor servers in the zone. Incoming reports are batched and broadcasted by razord-echo.
- Several minor fixes.
- Razor::Signature reoraganized. Moved hash() to ::String and renamed ::Signature to more appropriate, ::Catalogue. razoragents package doesn't require Net::Daemon and Persistence::Object::Simple anymore.
- razord can lookup multiple catalogues mounted at different places on the filesystem.





- razord heals corrupted objects.
- Added closest host discovery to Razor::Client. See razoragents/Changes for details.
- Added support for reading config options from /etc/razord.conf (server configuration) and ~/razor.conf (client configuration).
 Moved config code to Razor::Config
- Added protocol version to razord greeting

0.24 (first version of pretty good release, may 22, 2000)

• First public release.

0.20 (first version of peer review release, may 19, 1999)

• First peer-review release.

Generated on Tue Dec 4 07:02:43 2001 razor homepage



razor agents 1.17

What is Vipul's Razor?

Vipul's Razor is a distributed, collaborative, spam detection and filtering network. Razor establishes a distributed and constantly updating catalogue of spam in propagation. This catalogue is used by clients to filter out known spam. On receiving a spam, a Razor Reporting Agent (run by an end-user or a troll box) calculates and submits a 20-character unique identification of the spam (a SHA Digest) to its closest Razor Catalogue Server. The Catalogue Server echos this signature to other trusted servers after storing it in its database. Prior to manual processing or transport-level reception, Razor Filtering Agents (end-users and MTAs) check their incoming mail against a Catalogue Server and filter out or deny transport in case of a signature match. Catalogued spam, once identified and reported by a Reporting Agent, can be blocked out by the rest of the Filtering Agents on the network.

What should I download?

Fetch the latest versions of <u>razor-agents</u> and <u>razor-agents-sdk</u> packages. The SDK bundles all Perl modules required by razor-agents and should be installed before razor-agents. Required modules (<u>Net::Ping, Net::DNS, Time::HiRes, Digest::SHA1</u> and <u>Mail::Internet</u>) are also available from CPAN.

How do I install razor-agents?

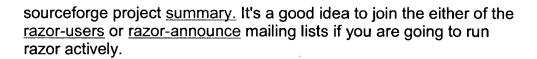
Both razor-agents and razor-agents-sdk are installed in the standard perl way: perl Makefile.PL; make; make test; su; make install. Once razor-agents is installed see manpages of razor-check and <a href="razor-chec

What has changed since last release?

The current release of Vipul's Razor is Release 1. Release 1 adds distributed functionality and fixes some long pending issues. For more information on changes see the change logs for <u>razor-agents</u> and <u>razord</u>. If you are still running razor agents from the <u>pretty good</u> release (0.24), you are strongly advised to upgrade.

Where is Razor developed?

Vipul's Razor is developed and hosted on sourceforge. Here's the



Where's all the documentation?

The following manpages are bundled in the razor-agents package, and also available online: razor-check(1), razor-check(1), razor-conf(5), Razor::Client(3)

Also see the <u>razor-users mailing list archive</u> for operational details and future development plans.

How can I help?

Razor's efficacy is proportional to the number of people using it. That is, more spam reports translate into fewer spams in your mailbox. If you are a Razor user, please ask your peers to use it too. If you are an ISP, you can setup a troll address (see <u>razor-report</u> manpage) where your users can bounce the spam they recieve, and integrate razor into your webmail system. If you are developer, you can help by contributing instructions and code to hookup razor agents with mail systems of your choice. And, finally, you can help by contributing code (see the <u>task list</u> for what remains to be done).

License

Copyright © 1999-2001, Vipul Ved Prakash. This software is distributed under the Artistic License (see the ARTISTIC file).





razor-check - Razor Filtering Agent

Synopsis

```
razor-check [-s -d -H] file_with_mail_in_rfc822_format
$ cat mail | razor-check
$ razor-check ./mail
$ razor-check -d ./mail
```

Description

razor-check checks a mail against the distributed Razor Catalogue. It should be invoked before the mail is delivered or processed by a human. razor-check terminates with exit value 0 if the signature for the mail is catalogued on the server (spam) or 1 if the mail is not catalogued by the server (not a spam). razor-check should be invoked against every incoming mail by mail processors (like procmail) or MTAs (like sendmail). An alternate method would be to call razor-check from cron, at regular intervals, to identify and mark spam in queued mailboxes.

Usage

razor-check is usually run by piping the contents of the mail to it, or by providing the name of the file that contains the mail message to be checked as the last argument. razor-check takes the following arguments:

-S

Simulate a check. Do everything except talk to the server.

-d | --verbose

Print debugging information.

-H

Compute and print the SHA hash of the mail contents. And then exit.

-M | --mbox

Accept a mailbox name on the command line and check every mail in the mailbox against the database. razor-check will print out the serial number of every mail found in the Catalogue. When combined with -n it will also print the serial numbers, prefixed with a -, of mails that were not found in the Catalogue. Usage:

```
razor-check -M ~/Mail/possible-spam
```

-S | --sig

Accept a list of pre-computed (with -H) signatures on the command line, instead of computing one from mail content. When called with --sig, razor-check will not attempt to read from STDIN. Usage:

```
razor-check -d -S a8a3d545adb73f9733675571ffeaf10cba87745b
```

-conf=filename

The path of the config file. By default the path is ~/razor.conf, an alternate path can be specified with this option. See razor.conf(5) manpage for various configuration options.

-h

Print a usage message and exit.

-V

Print the version number and exit.

Recipes

razor-check is usually invoked from procmail(1). Here are some common ways of using it with procmail:

To change the Subject header if mail is spam:

```
:0 Wc
| razor-check
:0 Waf
| formail -i "Subject: Razor Warning: SPAM/UBE/UCE"
```

To add a X-Razor-Warning header to spam:

```
:0 Wc
| razor-check
:0 Waf
```

```
| formail -A "X-Razor-Warning: SPAM."
```

To file spam in a mailbox

```
:0 Wc
| razor-check
:0 Wa
/home/foo/Mail/razor-caught
```

Author

Vipul Ved Prakash, <mail@vipul.net>

See also

razor.conf(5), razor-report(1), razor-discover(1)

License

This is free software, distributed under the **ARTISTIC** license.

Generated on Tue Dec 4 07:02:41 2001 razor homepage









Changes - razor-agent

1.17 (december 01, 2001)

Removed the ``Persistence::Object::Simple" dependency from test.pl

1.16 (november 27, 2001)

- Applied Nick Mossie's patch to fix --mbox option in Razor::Agents.
- Added support for split zones in Razor::Config::new(). Razor zone is now split into subzones, 'report' and 'check' to control the latency of report propagation through distributed servers. Servers listed in the reports zone accept checks and those in the check zone accept check request. It is important that users upgrade to this version otherwise their reports might be lost (as they will go to the nearest check servers, instead of a report server).

1.13 (november 13, 2001)

 Razor::Client/Agent don't exit() or die(); they cascade the error back to the caller through Razor::Errorhandler::errstr()

1.12 (november 7, 2001)

- Razor::Client returns exit values >1 for different errors. These are described in Razor::Client(5) manpage.
- Razor agents now use Getopt::Long instead of -s, to keep older versions of procmail happy. All old options are valid and retain their meaning.
- Moved command line parsing and common routines from razor-check/report to Razor::Agents.
- razor-check and razor-report can now work on mailboxes. A mailbox (in mbox forma can be provided on the command line with the --mbox option. razor-check/report w print the serial numbers of mails that were found in the catalogue.
- --sig option takes multiple signatures on the command line.
- All required modules are now bundled in razor-agents-sdk. Released version 1.00 of SDK.
- All modules use strict.

- Added support for checking and reporting precomputed signatures to razor-check an razor-report, through the -sig option.
- Added support for -H in razor-report.
- Razor::Signature reoraganized. Moved hash() to ::String and renamed ::Signature to more appropriate, ::Catalogue. razor-agents package doesn't require Net::Daemon a Persistence::Object::Simple anymore.
- Fixed a hairy bug in Razor::Client; it was too hairy to describe here.

1.06 (october 20, 2001)

- razor-check & razor-report now return correct exit values.
- Some small fixes to POD documentation.

1.05 (october 19, 2001)

• 1.04 was broken.

1.04 (first version of release 1, october 18, 2001)

Added closest host discovery to Razor::Client. Here's how it works: All razor servers that belong in a ``zone" (razor.vipul.net, by default) are named a.zone, b.zone, etc The client does a nslookup on a.zone, then b.zone and so on till it finds a domain th has a CNAME ``list.terminator". Client sends out TCP pings to echo ports of these ho and computes round trip time using Time::HiRes. The list of available servers is sortiosed on RTTs and the closest server is chosen.

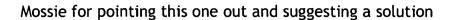
Rediscovery happens after rediscovery_wait amount of seconds, which is 7 (+-1) day by default. It can be forced with <razor-client -d> command.

Client moves from closest host to the next closest host, if:

- The server is not responding.
- The server has been upgraded and client doesn't understand the new protoc
- The server is broken, gives spurious answers.

The next closest host remains the chosen host till rediscovery. This takes care of server/network outages, semantically broken servers, and transition from old to nev protocols. Thanks to Peter J. Holzer for suggesting the use of a host naming scheme server discovery.

• A fix to Razor::Signature to workaround procmail munging of mail. Thanks to Nick



- Added a debug mode to Razor::Client
- Moved client API to Razor::Client
- Added support for reading config options from /etc/razord.conf (server configuratio and ~/razor.conf (client configuration). Moved config code to Razor::Config

0.24 (first version of pretty good release, may 22, 2000)

• First public release.

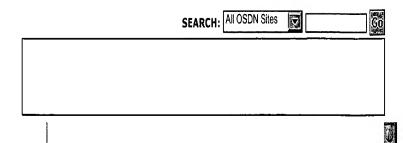
0.20 (first version of peer review release, may 19, 1999)

• First peer-review release.

Generated on Tue Dec 4 07:02:43 2001 razor homepage

OSDN: Open Source Development Network: Newsletters - Jobs





Status:

Project: Vipul's Razor

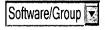
NOT LOGGED

IN Summary

Login via SSL № New User via SSL ☎

Summary | Admin | Home Page | Tracker | Bugs | Support | Patches | Lists | Tasks | Docs | Surveys | News | Files |

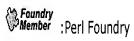
Search



Require All Words



Vipul's Razor is a distributed, collaborative, spam detection and filtering network. The primary focus of the system is to identify and disable an email spam before its injection and processing is complete.



Developer Info

Project
Admins:
hackworth

Project: Vipul's Razor

Project Summary 15

Project Admin №

Software

Software Map Software Map New Releases Software Mirrors Software Software Software Software Software Software Map New York Software New York N

Code Snippet
Library

- Development Status: 5 Production/Stable
- Environment: Console (Text Based), No Input/Output (Daemon)
- Intended Audience: Developers, End Users/Desktop, System Administrators
- License: Artistic License
- Natural Language: English
- Operating System: POSIXProgramming Language: Perl
- Topic: Filters, File Sharing

Registered: 2000-03-24 13:57

Activity Percentile (last week): 98.1142%

View project activity statistics

SourceForge.net

Site Docs S Work for

Latest File Releases

Developers:

2 [View Members]

Download

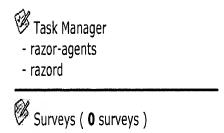
Download

Download

Download

Download

Sourceforge! S Discussion Forums	Package	Version	Date	Notes / Monitor	Downlo
Project Help Wanted Top Projects Site Status Contact Support	razor-obsolete-don't-download	0.24	May 21, 2000	9 - 19	Downlo
	razor-agents	1.17	December 1, 2001	9 - B	Downlo
	razor-agents-sdk	1.00	October 21, 2001	9 - B	Downlo
	rsync-patch	2.4.6	October 21, 2001	9 - 🗵	Downlo
Compile Farm ₪	•	[View ALL Project Files]			
SourceForge.net Foundries About Foundries	Public Areas	Latest News			
Clustering EDistributed Computing EDISTRIBUTED LINUX ON Large Systems EDISTRIBUTED LINUX Drivers EDIST	Project Home Page	1st public release of Vipul's Razor, spam filtering network. hackworth - 2000-05-21 10:49 (0 Comments) [Read More/Comment]			
	- Bugs (0 open / 1 total) Bug Tracking System - Support Requests (0 open / 0 total) Tech Support Tracking System - Patches (1 open / 1 total) Patch Tracking System - Feature Requests (0 open / 0 total) Feature Request Tracking System DocManager: Project	[News archive] [Submit News]			
	Documentation Mailing Lists (3 public mailing	-			
	lists)	_			





Show Source

© Copyright 2001 - **OSDN** Open Source Development Network, All Rights Reserved **About Sourceforge.net** • **About OSDN** • **Privacy Statement** • **Terms of Use** • **Advertise** • **Contact Us**